

Adapt the Words, Freeze the Model: LLM Prompt Evolution for Domain-Shift-Robust Segmentation

Olga Volkova^{1,2} Nikita Glazkov^{1,3} Konstantin Pchelin^{1,4} Iaroslav Bespalov¹

Evgeny Baraboshkin⁵ Denis Orlov⁵ Dmitry Dylov^{1,3}

¹AXXX ²Center for Petroleum Science and Engineering

³Applied AI Institute ⁴Institute of Complex Systems Mathematical Research MSU ⁵Digital Petroleum

{volkova,glazkov,pchelin,bespalov}@axxx.tech

baraboshkin-evgenij@ya.ru odm84@yandex.ru dylov@axxx.tech

Abstract

In specialist industrial imagery the dominant cost is maintenance under domain shift: labeling and fine-tuning must be redone on every new field, and proprietary imagery often cannot leave the operator’s environment. For geological drill-core fracture segmentation we release CoreFrac, to our knowledge the first open pixel-level benchmark on drill-core photographs: 200 expert-annotated full-resolution images with an expert-agreement ceiling (soft-F1 0.86). We retarget a frozen concept segmentor (SAM 3) by evolving only its natural-language prompt—no gradient updates, just a small labeled probe. Under a source-grouped protocol the evolved prompt reaches 0.545 positives-only soft-F1, beating rank-8 LoRA at the same 96-patch cap (0.487; paired +0.058, 95% CI [+0.020, +0.094]) and the best of eight expert prompts by 2.7 $\times$. Under a controlled domain shift, re-evolving on a small probe lifts a 0.622 $\rightarrow$ 0.409 drop back to 0.534. Its masks distill into a small on-premises U-Net (7.8 M parameters, 3.8 ms/patch) that retains $\sim$ 92% of the teacher’s positives-only soft-F1 (0.501) at $\approx$ 1700 $\times$ lower latency than *unbatched* SAM 3—a cheaper, leak-free deployment. Code and data are released.

1 Introduction

Geological core analysis underpins reservoir characterization in oil and gas and ore assessment in mining. Specialists trace fractures on high-resolution photographs of drill core to produce the binary masks that feed permeability, geomechanical, and rock-quality models. This manual logging is slow and expensive, which motivates automation.

The default recipe—annotate cores, train a U-Net (Ronneberger et al., 2015), deploy—breaks down under maintenance: lithology, lighting, resolution, and fracture appearance vary across fields and rigs, so a model degrades on the next campaign, and recovery means re-annotating and retraining—the very cost automation was meant to remove. A

second, often unstated constraint compounds it: operators are frequently forbidden from sending proprietary core imagery to cloud models, so the deployed component must run on-premises and cheaply (we call this *leak-free*).

We address both with a three-stage pipeline whose only adapted component is natural language. Open-vocabulary segmentors such as SAM 3 (Carion et al., 2025) mask every instance of a noun phrase, so a domain change reduces in principle to changing the phrase—no labels, no training. Because hand-crafting effective phrasings is hard and foundation models underperform out of the box on specialist imagery (Mazurowski et al., 2023), we automate it: an LLM mutates candidate prompts and keeps those a frozen segmentor scores highest against expert masks, in the spirit of Evo-Prompt (Guo et al., 2024) but with a *perceptual* fitness and a *segmentor* as the consumer.

A frozen model good enough to be useful is valuable even below human accuracy: we use it as an *annotation aid*—it pre-labels fractures that experts merely correct—and *distill* (Hinton et al., 2015) its masks into a small U-Net that runs on-premises with no large model in the loop.

The pipeline targets a real deployment—a rock-image analysis tool for lithological description in oil-and-gas and mining—so we foreground system concerns: latency, maintenance under domain shift, human-in-the-loop labeling, and on-premises data locality.

This paper makes three contributions, framed around maintenance:

1. **A benchmark for maintaining language-prompted segmentation.** We release CoreFrac (§3): to our knowledge the first open *pixel-level* benchmark for fracture segmentation on drill-core *photographs*—200 full-resolution expert-annotated images, a patch protocol that scores empty and positive re-

gions separately, an explicit positive-class taxonomy, and a human inter-annotator ceiling (soft-F1 0.86 ± 0.12).

2. **Label-efficient adaptation by language rather than retraining.** An automatically evolved natural-language prompt retargets a frozen SAM 3 with no gradient updates and no weight adaptation—using only a small labeled scoring probe rather than a re-annotated training corpus—turning annotate-and-retrain maintenance into a cheaper, auditable language loop. It beats matched-cap rank-8 LoRA (0.545 vs. 0.487) and, by a wide margin, the best of eight expert prompts; under domain shift it recovers most of the induced drop by re-prompting alone, and the same loop transfers and re-adapts on an external source (§6.2–§6.4).
3. **The loop yields operator value and an on-prem artifact.** The frozen model shows a preliminary expert-annotation speed-up and—the well-measured leg—distills its masks into a small on-premises U-Net (7.8 M params, 3.8 ms/patch) that retains $\sim 92\%$ of the teacher’s positives-only soft-F1 (0.501 vs. 0.545) with no manual masks in its targets (§6.5–§7).

2 Related Work

The Segment Anything line established promptable segmentation (Kirillov et al., 2023), added video memory in SAM 2 (Ravi et al., 2025), and concept segmentation from noun phrases in SAM 3 (Carion et al., 2025). Phrase-grounded detectors (Grounding DINO (Liu et al., 2024), Grounded-SAM (Ren et al., 2024)) and language-reasoning segmenters (LISA (Lai et al., 2024)) offer other text-to-mask routes. All degrade on specialist domains (Mazurowski et al., 2023), which motivates leaving the backbone frozen and adapting elsewhere.

For prompt optimization, APE (Zhou et al., 2023) and OPRO (Yang et al., 2024) use LLMs to refine instructions; EvoPrompt (Guo et al., 2024) and Promptbreeder (Fernando et al., 2023) evolve textual prompts for textual tasks. FunSearch (Romera-Paredes et al., 2024) and AlphaEvolve (Novikov et al., 2025) place an LLM in an evolutionary loop with an automatic evaluator, with MAP-Elites (Mouret and Clune, 2015) keeping a diverse elite archive. Our novelty is to optimize

a *textual* prompt whose consumer is a *segmentor*, with a *perceptual* fitness, on the open-source GigaEvo framework (Khrulkov et al., 2025), and to use the loop as a maintenance mechanism.

Large models increasingly bootstrap labels that smaller models learn from; SAM itself was built with a model-in-the-loop data engine (Kirillov et al., 2023). In NLP this LLM-as-annotator paradigm trains task-specific models from generated labels with only a small human-labeled pool, often matching or beating the teacher (Zhang et al., 2023; Tan et al., 2024); distillation for low-latency deployment is likewise an established Industry-Track concern (Brown et al., 2023). We combine foundation-model pre-labeling with expert correction and knowledge distillation (Hinton et al., 2015) into a compact deployable model—instantiating that paradigm in a vision setting, where the “annotator” is a frozen concept segmentor steered by an evolved phrase rather than an LLM, and the consumer is a small on-premises network.

Turning to data, public crack resources target pavements (CRACK500/FPHBN (Yang et al., 2020), DeepCrack (Zou et al., 2019)) or geological *outcrops* (GeoCrack (Yaqoob et al., 2024)); rock-CT fracture segmentation uses supervised U-Nets (Wu et al., 2024). For *drill core*, the closest work segments unwrapped-core images on proprietary data (Alzubaidi et al., 2022), and public drill-core fracture releases provide annotated PDFs/spreadsheets rather than pixel masks (Jones, 2021). No open pixel-level benchmark targets drill-core *photographs*; CoreFrac fills that gap. We use CRACK500 only as an out-of-domain transfer probe (§6.4), not a same-task substitute.

3 The CoreFrac Benchmark

We release 200 full-resolution drill-core photographs with expert pixel masks under an open license—to our knowledge the first open pixel-level benchmark on drill-core photographs (§2). Data and code are available at <https://github.com/Lelevolkova/CoreFrac>. We ship full resolution so others can choose their own tiling, and derive a fixed patch benchmark from it. We state our binary mapping explicitly (it ships with the data): positive = an open, healed, or mineralized fracture, fissure, or vein on the core surface; negative = bedding planes, drilling scratches, grain texture, core-tray edges, depth labels, and intact rock. Figure 1 shows full-resolution cores spanning this diversity; the

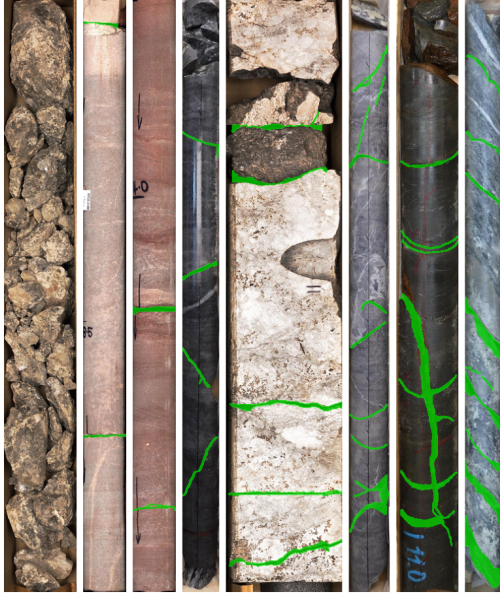


Figure 1: Diversity of the CoreFrac source data: full-resolution drill-core columns with expert masks overlaid (green), spanning fracturing degree (intact to heavily fractured), rock type and color, dry vs. wet surfaces, and in-frame acquisition noise such as depth labels, orientation arrows, and core-tray edges.

patch-level taxonomy is in Figure 3.

From these images we curate 450 diversity-sampled patches: 360 positive (at least one fracture) and 90 empty. Empty patches are first-class, since a deployable system must not hallucinate fractures on intact rock. Headline numbers use a *source-grouped* split: 99/179/172 patches from 18/25/23 train/validation/test cores, with zero source-core overlap (Table 5). From the 99 training patches we fix a 96-patch adaptation cap shared by every method; the other three sit outside it. All model selection uses source-disjoint validation groups.

We binarize prediction P and ground truth G at a fixed threshold and resize to annotation resolution. Strict pixel Dice = $2|P \cap G|/(|P| + |G|)$ scores a one-pixel-wide crack displaced by a single pixel as 0 although it is visually correct, so as primary we report a soft tolerance-F1: a predicted pixel counts as correct within τ pixels of a ground-truth crack pixel (and vice versa), $\tau = 2$ px to match expert stroke width—standard for thin-structure crack benchmarks (Yang et al., 2020; Zou et al., 2019). Strict Dice is reported alongside as a conservative reference, so no gain is a tolerance artifact. An empty prediction on an empty patch scores 1.0; since empties score trivially high, positives-only is primary and the all-patch mean secondary (with

IoU, precision/recall, success rate, and latency reported alongside).

To bound achievable accuracy we define an expert-agreement ceiling: pairwise agreement between two qualified geologists on a shared 20-image subset under the same metric. They agree on crack presence in all 20 images and reach 0.862 ± 0.122 soft-F1 (0.751 ± 0.101 strict, 0.611 IoU); disagreement is confined to crack extent, not existence—the thin-structure effect the soft metric absorbs. No system approaches this ceiling (the champion reaches $\sim 63\%$).

4 Method: Evolve, Annotate, Distill

The pipeline has three stages (Figure 2), presented as built and deployed: a frozen segmentor adapted by language (Stage 1), used to accelerate annotation (Stage 2), then distilled into a small deployable model (Stage 3)—replacing the annotate-and-retrain maintenance loop with a language-driven one that needs no per-domain gradient updates.

Stage 1 – Evolve (language-only adaptation). A prompt p (one English phrase, ≤ 180 chars, e.g. “cracks in drill core”) is passed to a frozen SAM 3 as a concept query; returned masks are unioned above a fixed confidence, resized, and binarized—no weights, thresholds, or metric change. We optimize p with MAP-Elites (Mouret and Clune, 2015) on GigaEvo (Khurlov et al., 2025): an LLM (Qwen3-235B) mutates archived prompts and the frozen pipeline scores each by soft tolerance-F1 on the train probe. Evaluation uses no LLM, so fitness is cheap, deterministic, and auditable. The direct alternative to this language interface is parameter-efficient weight adaptation of the same frozen segmentor; we therefore treat matched-cap LoRA as a first-class baseline (§5).

Stage 2 – Annotate (human-in-the-loop). Even a sub-human model is useful if it removes blank-page cost (annotating from scratch). We run the frozen, prompt-adapted SAM 3 over the unlabeled corpus to produce *pre-labels* that experts accept, reject, or edit. Measuring the saving fairly needs a counterbalanced protocol (matched from-scratch vs. correction sets, no image seen twice by one annotator); we report only exploratory self-timed minutes per image (§6.5). This saving is the payoff of Stage 1 independent of whether the model reaches human accuracy.

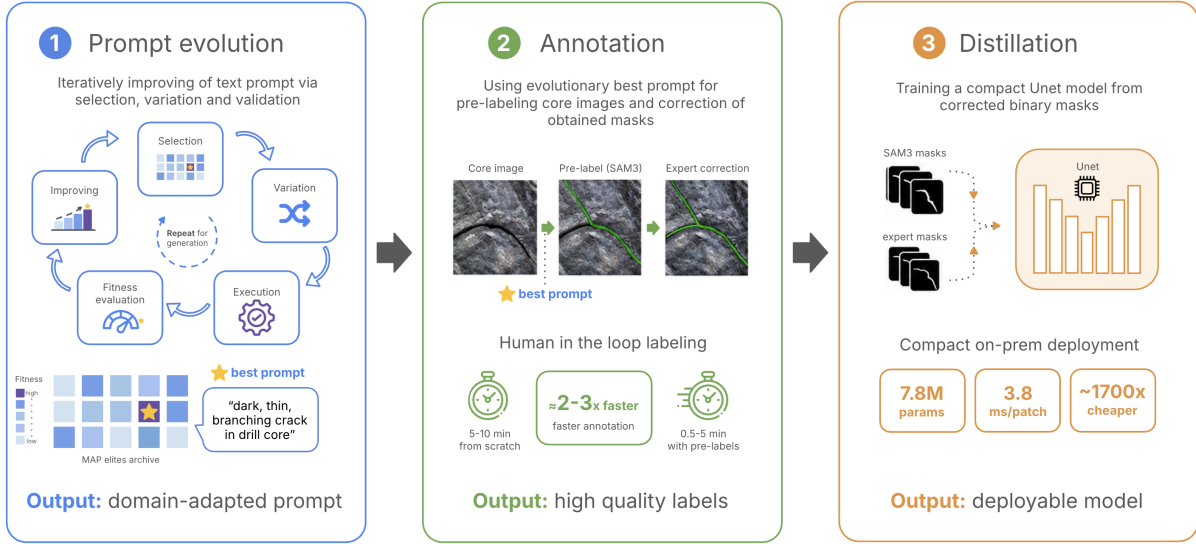


Figure 2: The CoreFrac pipeline. **(1) Evolve** (language-only): an LLM evolves a text prompt for a frozen SAM 3 against a small labeled probe. **(2) Annotate** (human-in-the-loop): the prompt-adapted model pre-labels new cores that geologists only correct, and re-prompting alone recovers a controlled domain shift. **(3) Distill**: the masks distill into a compact on-prem U-Net (7.76 M params, 3.8 ms/patch). No gradient updates touch the frozen segmentor.

Stage 3 – Distill (cheap, leak-free deployment). SAM 3 is open-weight but large and GPU-costly to serve, so we distill (Hinton et al., 2015) the (pseudo- or human-corrected) masks into a small U-Net (7.76 M parameters) that needs no foundation model at inference: cheaper on-prem inference and—no hosted large model required—a deployment that satisfies policies forbidding proprietary imagery from leaving the operator’s environment.

5 Experimental Setup

Segmentor and optimizer. SAM 3 (official open-weight facebookresearch/sam3 release, transformers 5.0.0 Sam3Model), confidence 0.5, full-image and patch-stitch inference. Mutation LLM Qwen3-235B (Yang et al., 2025) on a local endpoint; budget 2000 candidates (an initial 1000-candidate run extended to 2000); concurrency 4 (one GPU each); fitness = soft tolerance-F1 on a fixed 64-patch subset of the 96-patch train cap. Headline comparisons use three independent seeds; values are mean $\pm$ SD, and paired intervals resample the 23 test cores as clusters (Appendix A).

Baselines and conditions. (i) SAM 3 seed and expert-written prompts (Appendix A); (ii) **rank-8 LoRA** on SAM 3 (Hu et al., 2022) under the same 96-patch cap, preprocessing, source-grouped validation groups, and test cores—learning rate and checkpoint use only grouped validation; (iii) fine-tuned U-Net on full manual labels (supervised up-

per bound); (iv) **distilled U-Net** on SAM 3 pseudo-labels; (v) zero-shot VLM sweep (Qwen3-VL (Bai et al., 2025) and others; polygons/grids rasterized to masks); (vi) open-weight on-prem zero-shot segmentors—OWLv2→SAM (Minderer et al., 2023), CLIPSeg (Lüddecke and Ecker, 2022), and Florence-2 (Xiao et al., 2024)—whose shared failure modes are shown in Fig. 4. Prompt fitness uses 64 of the 96-cap patches, while LoRA is allowed all 96, so we call the cap matched, not realized label access identical. Hardware: 4× GTX 1080 Ti (LoRA excepted, Appendix A).

Deployment. The distilled “core-fracturing” stage was studied as a standalone crack-segmentation module for a production rock-image analysis tool. This is the work’s single deployment claim: an offline evaluation that gated integration under one industrial acquisition setting, not unattended or multi-site production use. An online evaluation (correction-acceptance rates, end-to-end throughput) is the natural next step.

6 Results

6.1 Optimization trajectory

We trace the best-so-far prompt of the soft-optimized run on the original ungrouped split (full frontier in Appendix A, Table 3). The selection objective rises 0.156→0.555, but transfer is *not* monotonic: verbose mid-run prompts over-

fit the fitness sample (test ≈ 0), while the concise champion generalizes, lifting positives-only soft-F1 0.013 $\rightarrow$ 0.563 (strict 0.444) and keeping 24/35 empty patches clean. Re-evolving from scratch on the grouped pool reproduces the effect (0.545 on the grouped test, §6.2) with champion “dark thin branching fracture in drill core”—a light lexical edit of the ungrouped phrase (“crack” $\rightarrow$ “fracture”, commas dropped); intact-rock false positives remain the main weakness.

6.2 System comparison

Table 1 reports the main comparison. Under the source-grouped protocol the evolved prompt reaches 0.545 ± 0.011 positives-only soft-F1 against 0.487 ± 0.016 for rank-8 LoRA at the same 96-patch cap (paired $+0.058$, 95% CI $[+0.020, +0.094]$, 23 test cores as clusters), with fewer empty-patch false positives (9.7 vs. 14.0/38) and no domain-specific weight artifact. It stays within 0.034 of the fully supervised U-Net (0.579 ± 0.013 ; 11.3/38 empty FPs), and the zero-annotation student retains 87% of that (0.501 ± 0.017). Hand-written prompts are far behind—the best of eight reaches 0.212 where the evolved phrase scores 0.563 on the ungrouped split ($2.7\times$)—so search justifies itself against a domain expert.

6.3 Maintenance under domain shift

As a first, controlled test we induce a shift *within* CoreFrac by splitting on core brightness—lighter healed-fracture cores (domain A) vs. darker open-fracture cores (domain B), source-aware so no core spans domains. This internal proxy isolates a visual-signature shift while holding acquisition and annotation fixed; it is a different split from the headline grouped protocol, and it is *not* a field-to-field, rig, or operator change (Limitations). We evolve on domain A, observe the drop on B, then re-evolve on a small B probe—no gradient updates—and measure the recovery (Table 2).

The drop is real and the recovery label-efficient: the A-evolved prompt falls 0.622 $\rightarrow$ 0.409 positives-only soft-F1 (0.482 $\rightarrow$ 0.328 strict) on A $\rightarrow$ B, and re-evolving *only the prompt* on a small B probe lifts B-test to 0.534 (0.432 strict)—an 85.8% recovery of in-domain A, no gradient updates or corpus re-annotation. The generic seed scores 0.0 on both, so the lever is prompt quality, not the segmentor; the effect replicates across three seeds (recovery/decay = 0.90 ± 0.23), seeds 2–3 converging on

the *identical* prompt “dark, thin, branching crack in drill core”. The recovery carries a precision cost: the B-recovered prompt raises empty-patch false positives (13/24 vs. 3/24) and under an iso-FP sweep only dominates above a ≈ 9 –10/24 budget, so the two prompts are selectable high-recall/high-precision operating points (Appendix A). The expensive alternative—re-annotating domain B and retraining the U-Net—reaches 0.647 soft (0.515 strict); re-prompting recovers $\sim 83\%$ of that from a small probe alone.

6.4 External-domain transfer (CRACK500)

To probe a genuinely external source we evaluate on CRACK500 (Yang et al., 2020), 1,124 held-out smartphone pavement-crack patches—a different sensor, scene, operator, and crack morphology (strict Dice; Table 6). The drill-core-evolved prompt transfers *as-is* (0.460 Dice, matching a generic “crack” prompt’s 0.457), while an over-specified phrase collapses (0.135)—the same brittleness seen in-domain. The identical language-only loop on a tiny 15-patch probe lifts Dice to 0.471 at higher precision, independently discovering a domain-appropriate negative clause (“... not shadow or edge”). Pavement cracks are high-contrast, so zero-shot transfer is already strong and re-prompting headroom small; we read this as the pipeline *transferring and re-adapting on an external source*, not a like-for-like accuracy comparison (CRACK500 Dice is not comparable to our thin-hairline soft tolerance-F1).

6.5 Annotation acceleration

Stage 2’s payoff is wall-clock time, not accuracy. A preliminary deployment measurement supports a speed-up: geologists spend 5–10 min per core from scratch versus 0.5–5 min correcting SAM 3 pre-labels (upper end on heavily fractured sections), a ~ 2 – $3\times$ median, with most effort spent deleting structured false positives rather than tracing missed cracks. This is preliminary, self-reported evidence; the counterbalanced protocol (§4) was not run.

6.6 Distillation and label efficiency

The deployable artifact is a small U-Net distilled from the frozen segmentor’s pseudo-labels alone—no human labels touch the targets—and it retains $\sim 92\%$ of the teacher’s positive accuracy (soft-F1 0.501 vs. 0.545; strict Dice 0.389 vs. 0.429) and 87% of full supervision (0.579) at 7.8 M parameters and 3.8 ms/patch ($\sim 1700\times$ cheaper than un-

System	Output	Trained	Supervision	soft-F1 (primary)		Dice	
				pos	all	pos (strict)	Lat. (s)
<i>Promptable foundation segmentors (frozen)</i>							
SAM 3, seed prompt	native	no	no	0.013	0.210	0.012	6.5
SAM 3, best human prompt	native	no	probe	0.212	0.363	0.168	6.5
SAM 3, evolved prompt (ours)	native	no	probe	0.545	0.589	0.429	6.5
SAM 3, rank-8 LoRA (matched cap)	native	yes	probe	0.487	0.519	0.371	6.5
OWLv2 → SAM (open-vocab)	native	no	no	0.211	0.266	0.142	~1.0
<i>Zero-shot VLM / open-vocab segmentors</i>							
Best zero-shot VLM (Qwen3-VL)	polygon	no	no	0.103	0.191	0.048	—
CLIPSeg (text → mask)	dense	no	probe	0.152	0.236	0.114	~0.1
Florence-2 (referring seg.)	polygon	no	no	0.150	0.120	0.118	~8
<i>Deployable supervised models</i>							
Distilled U-Net (SAM 3 pseudo)	native	yes	pseudo	0.501	0.493	0.389	0.004
Fine-tuned U-Net (full labels)	native	yes	full	0.579	0.606	0.468	0.004
<i>Human inter-annotator ceiling</i>	—	—	—	0.862	—	0.751	—

Table 1: System comparison on the CoreFrac test split. Primary metric: soft tolerance-F1 ($\tau = 2$), positives-only (**pos**) and all-patch (**all**); strict pixel Dice alongside. Adapted systems are three-seed means. **Output**: *native* instance masks, traced *polygon*, or *dense* heatmap. **Supervision**: *no*; *probe* (small scoring set); *pseudo* (SAM 3 pseudo-labels); or *full* (manual masks). Latency per 256×256 patch on one GTX 1080 Ti. The human ceiling (§3) bounds every system; baseline protocols, LoRA settings, and uncertainty estimation in Appendix A.

Prompt / model	A-test soft (strict)	B-test soft (strict)
Seed prompt	0.000 (0.000)	0.000 (0.000)
Evolved on A (language)	0.622 (0.482)	0.409 (0.328) ↓
Re-evolved on B (language)	0.638 (0.491)	0.534 (0.432) ↑
Re-annotate + retrain U-Net (B)	—	0.647 (0.515)

Table 2: Maintenance under domain shift (positives-only soft tolerance-F1, strict Dice in parentheses; representative seed). A prompt evolved on A degrades on B; re-evolving on a small B probe recovers most of the drop (no gradient updates, no corpus re-annotation). Last row: the expensive re-annotate-and-retrain baseline. Across three seeds recovery/decay = 0.90 ± 0.23 ; per-seed and iso-FP analysis in Appendix A.

batched SAM 3, on-prem; Table 1). The measured cost is empty-patch false positives, which it amplifies rather than inherits ($9.7 \rightarrow 20.3/38$), pulling all-patch soft-F1 to 0.493—below its own positives-only score, since empty-patch errors now dominate. Expert-correction and equal-annotation-minute controls were not run (thresholds calibrated on validation, never test; Appendix A).

6.7 Cost of adaptation and inference

The industrial case in numbers (Appendix A, Table 4): per new domain we replace a re-annotated corpus with a tens-of-patches probe, trading ~ 0.05 supervised GPU-hours for ~ 7 (evolve) + 0.6 (distill). Both arms deploy the *same* 7.76 M-param U-Net (31.1 MB checkpoint, 1.9 GB inference

VRAM, 3.8 ms/patch), so the method trades adaptation compute for scarce expert time rather than claiming lower cost on every dimension.

7 Discussion

Read as a maintenance mechanism, the pipeline degrades under a controlled intra-dataset shift and recovers by re-prompting alone (§6.3): adaptation is an auditable language edit—a short English phrase rather than opaque weights—not an annotate-and-retrain cycle. The NLP contribution is this testable interface: domain-specific state is a short, inspectable, versionable phrase while the foundation model stays frozen. The matched-cap comparison tests that interface against weight adaptation and supports superiority over the tested rank-8 LoRA configuration, not over PEFT families generally. Evidence beyond SAM 3/CoreFrac is partial but positive: the evolved phrase raises CLIPSeg soft-F1 from 0.020 to 0.152 (a cross-model word-ing signal, not a full CLIPSeg-consumer validation), and CRACK500 re-evolution discovers “not shadow or edge” (+0.011 Dice). We present those as supporting signals and limit the central claim to SAM 3 on source-grouped CoreFrac and the tested rank-8 configuration.

The frozen model does not reach human accuracy; it can still accelerate expert annotation (§6.5). Distillation then yields a small on-prem model that keeps data in-house, at a measured empty-FP cost

(§6.6, §6.7).

Four lessons generalize past this dataset and, we expect, past this task: a square model input silently destroys extreme-aspect imagery unless tiling matches the training geometry; more descriptive prompts overfit a small probe rather than helping; the operative tokens turn out to be generic (“open”, “crack”) rather than geological; and default distillation hyperparameters break empty-region behavior until threshold, loss weighting, and checkpoint selection are made empty-aware (Appendix A).

Limitations

Headline numbers use the source-grouped split (99/179/172 patches, 18/25/23 cores, zero overlap). Its predecessor was leaky: 61 of 66 source cores appeared in more than one split and some train/test crops overlapped in source pixels. Relative to that split the global prompt moved from 0.563 to 0.545 (−0.018) and the patch-trained student from 0.558 to 0.501 (−0.057); because test composition also changed, these are not paired tests, but the larger student drop is why grouping was required. Annotation timing is exploratory and self-reported; a counterbalanced protocol and extra distillation controls were not run. Absolute accuracy is still modest—the evolved prompt reaches soft-F1_{pos} 0.545 (strict Dice 0.429) and false-positives on about 26% of intact grouped-test patches (9.7/38)—predominantly on *structured* non-rock content (vertical drilling/depth markings and core-tray borders) rather than at random, which practitioners can suppress by masking tray regions—so we position it as an annotation accelerator and distillation teacher rather than an autonomous labeler. False positives—not misses—are the chief deployment risk: the gain from re-prompting is operating-point dependent and reverses at the strictest empty-FP budget (§6.3). CoreFrac covers one industrial source and one task; the drill-core domain-shift study is a controlled intra-dataset split (core brightness / fracture type), distinct from the headline grouped protocol, rather than a field-to-field, rig, or operator change. CRACK500 (§6.4) shows limited procedure transfer on easier pavement cracks, not broad zero-shot prompt transfer. We compare against one rank-8 LoRA configuration at a 96-patch cap and do not claim superiority over other PEFT families, ranks, or budgets. The $\sim 1700\times$ inference saving is un-

batched SAM 3 vs. the small U-Net and would shrink against a batched/optimized foundation-model server. All imagery follows standard geological core-photography practice—individual core columns extracted from the core box and photographed in daylight, one core run per frame, with dry or lightly moistened surfaces; we have not evaluated robustness to artificial or UV lighting, wet-screen logging, in-situ borehole/televviewer imaging, or whole-tray multi-column photographs. The approach inherits SAM 3’s concept resolution: structures it cannot ground (the thinnest hairline cracks) are unreachable by any prompt, which also caps the quality of pre-labels and hence the distilled model. Evolved prompts need a small labeled probe and are stochastic (mitigated with multiple seeds) and may be brittle to SAM 3 version changes. Results are reported at $\tau = 2$ px; we do not sweep τ , though strict Dice ($\tau = 0$) is reported throughout as a lower bound. Distillation inherits systematic errors from pre-labels; expert correction was not isolated in the grouped study.

Ethics Statement

The drill-core imagery contains no personal data and is used with permission; we release an open, anonymized subset to enable reproducibility, and the deployed inference model is a small on-premises network so proprietary imagery need not leave the operator’s environment. The public repository contains the dataset, the expert masks, the split manifest and its provenance, the 96-patch adaptation ledger, the seed and evolved prompts, the evaluation, inference, and LoRA-training code, the cluster-bootstrap script, and the matched-cap LoRA configuration—enough to re-score any system under our protocol and to recompute the reported comparisons and intervals from the resulting per-patch scores. Training logs, prompt-search traces, model checkpoints, and per-patch prediction dumps are not released; SAM 3 weights are an external dependency cited to their official source under their license, not redistributed. Automated fracture logging is decision support: we do not advocate removing expert review from safety- or investment-critical interpretation, and we report success rate and failure cases so operators can calibrate trust.

Acknowledgments

We thank the geologists who annotated CoreFrac and the anonymous reviewers and area chair for comments that improved the evaluation protocol.

References

- Fatimah Alzubaidi, Harikrishnan Nalinakumar, Stuart R. Clark, Peyman Mostaghimi, and Ryan T. Armstrong. 2022. [Automatic fracture detection and characterization from unwrapped drill-core images using mask R-CNN](#). *Journal of Petroleum Science and Engineering*, 208:109471.
- Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhi-fang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025. [Qwen3-VL technical report](#). *CoRR*, abs/2511.21631.
- Nathan Brown, Ashton Williamson, Tahj Anderson, and Logan Lawrence. 2023. [Efficient transformer knowledge distillation: A performance review](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 54–65, Singapore. Association for Computational Linguistics.
- Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, and 19 others. 2025. [SAM 3: Segment anything with concepts](#). *CoRR*, abs/2511.16719.
- Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. 2023. [Promptbreeder: Self-referential self-improvement via prompt evolution](#). *CoRR*, abs/2309.16797.
- Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. 2024. [EvoPrompt: Connecting LLMs with evolutionary algorithms yields powerful prompt optimizers](#). In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. [Distilling the knowledge in a neural network](#). *CoRR*, abs/1503.02531.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *The Tenth International Conference on Learning Representations (ICLR)*.
- Clay Jones. 2021. [Utah FORGE well 16a\(78\)-32 core photos](#). DOE Geothermal Data Repository.
- Valentin Khrulkov, Andrey Galichin, Denis Bashkurov, Dmitry Vinichenko, Oleg Travkin, Roman Alferov, Andrey Kuznetsov, and Ivan Oseledets. 2025. [GigaEvo: An open source optimization framework powered by LLMs and evolution algorithms](#). *CoRR*, abs/2511.17592.
- Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. 2023. [Segment anything](#). In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003.
- Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. 2024. [LISA: reasoning segmentation via large language model](#). In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9579–9589.
- Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. 2024. [Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection](#). In *Computer Vision – ECCV 2024*, volume 15105 of *Lecture Notes in Computer Science*.
- Timo Lüddecke and Alexander Ecker. 2022. [Image segmentation using text and image prompts](#). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7086–7096.
- Maciej A. Mazurowski, Haoyu Dong, Hanxue Gu, Jichen Yang, Nicholas Konz, and Yixin Zhang. 2023. [Segment anything model for medical image analysis: An experimental study](#). *Medical Image Analysis*, 89:102918.
- Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. 2023. [Scaling open-vocabulary object detection](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Jean-Baptiste Mouret and Jeff Clune. 2015. [Illuminating search spaces by mapping elites](#). *CoRR*, abs/1504.04909.
- Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. 2025. [AlphaEvolve: A coding agent for scientific and algorithmic discovery](#). *CoRR*, abs/2506.13131.
- Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloé Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu,

- Ross B. Girshick, Piotr Dollár, and Christoph Feichtenhofer. 2025. [SAM 2: Segment anything in images and videos](#). In *The Thirteenth International Conference on Learning Representations (ICLR)*.
- Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kun-chang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. 2024. [Grounded SAM: assembling open-world models for diverse visual tasks](#). *CoRR*, abs/2401.14159.
- Bernardino Romera-Paredes, Mohammadamin Barekatin, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. 2024. [Mathematical discoveries from program search with large language models](#). *Nature*, 625(7995):468–475.
- Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. [U-Net: Convolutional networks for biomedical image segmentation](#). In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241.
- Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansoor Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. [Large language models for data annotation and synthesis: A survey](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 930–957, Miami, Florida, USA. Association for Computational Linguistics.
- Xiang Wu, Fei Wang, Xiaoqiu Zhang, Bohua Han, Qianru Liu, and Yonghao Zhang. 2024. [A deep learning approach for automatic segmentation of rock micro-fractures](#). *Engineering Geology*.
- Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. 2024. [Florence-2: Advancing a unified representation for a variety of vision tasks](#). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4818–4829.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *CoRR*, abs/2505.09388.
- Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2024. [Large language models as optimizers](#). In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Fan Yang, Lei Zhang, Sijia Yu, Danil Prokhorov, Xue Mei, and Haibin Ling. 2020. [Feature pyramid and hierarchical boosting network for pavement crack detection](#). *IEEE Transactions on Intelligent Transportation Systems*, 21(4):1525–1535.
- Mohammed Yaqoob, Mohammed Ishaq, Mohammed Yusuf Ansari, Venkata Ram Sagar Konagandla, Tamim Al Tamimi, Stefano Tavani, Amerigo Corradetti, and Thomas Daniel Seers. 2024. [GeoCrack: A high-resolution dataset for segmentation of fracture edges in geological outcrops](#). *Scientific Data*, 11:1318.
- Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023. [LLMaAA: Making large language models as active annotators](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13088–13103, Singapore. Association for Computational Linguistics.
- Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2023. [Large language models are human-level prompt engineers](#). In *The Eleventh International Conference on Learning Representations (ICLR)*.
- Qin Zou, Zheng Zhang, Qingquan Li, Xianbiao Qi, Qian Wang, and Song Wang. 2019. [DeepCrack: Learning hierarchical convolutional features for crack detection](#). *IEEE Transactions on Image Processing*, 28(3):1498–1512.

A Reproducibility details

Patch-level taxonomy. Figure 3 shows representative *patches* cropped from the full cores of Figure 1, ordered along the positive-class taxonomy from hairline to wide/healed fractures plus an intact (empty) patch. At patch scale the fractures are thin, low-contrast, and easily confused with bedding, grain texture, core-tray edges, and drilling artifacts—note the black depth-marker stroke on the intact patch, which a fracture detector must *not* segment.

Full prompt frontier (ungrouped run). Table 3 lists every best-so-far prompt of the original ungrouped run with its character count, soft train objective, and held-out test scores; the grouped revolution reports only its endpoint (0.545 ± 0.011 , §6.2). The champion is the *shortest* phrase on the frontier (41 characters); the elaborate mid-run prompts (up to 98 characters) raise the train objective but find no true positives on the held-out test—verbosity overfit the small probe. For reference, the prior-run champion (from the earlier strict run) “cracks in drill core” (20 characters), scored under the soft pipeline, reaches soft 0.547 / 0.608 (positives-only / all-patch), strict 0.425 positives-only, with 30/35 empties clean—competitive with the soft-run champion and cleaner on empties.

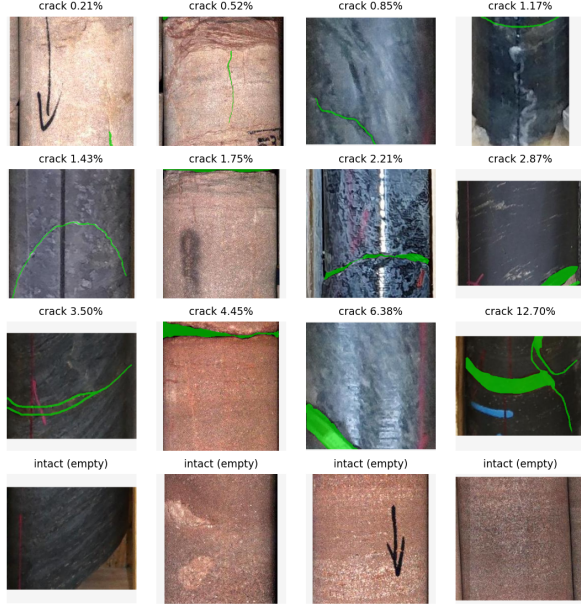


Figure 3: CoreFrac patches with expert masks overlaid (green), spanning the positive-class taxonomy from hairline to wide/healed fractures plus an intact (empty) patch. Percentages are the fraction of crack pixels in the patch.

Multi-seed reproducibility (train objective, ungrouped run). We re-run the full evolution as independent seeds with different stochasticity. A second seed (1000 mutants) re-derived the headline champion *verbatim*—“dark, thin, branching crack in drill core” at train soft-F1 0.5547 / strict 0.4722, matching the first seed to four decimals—and independently surfaced equally fit but *shorter* phrases: “open crack” (0.5557), “cracks in drill core, not bedding” (0.5564), and even a bare “crack” (0.5122). The operative tokens are *open/crack*; heavy geological specificity adds little—an Industry-relevant sign that the evolved prompt is short, cheap, and readable. These are train-objective figures on the 64-patch sample; the strongest robustness signal is that an independent seed re-derived the headline champion *verbatim* on this objective, and per-seed *test* variance is reported for the maintenance study (§6.3, three seeds).

Strict-metric precursor (why the soft metric). An earlier evolution run scored under strict pixel Dice (tol= 0) motivated the switch to the soft tolerance-F1. Its frontier climbs from train Dice 0.156 (seed) to 0.451 (champion “cracks in drill core”), and on the held-out test the champion (scored under *strict* Dice at this run’s own operating point) reaches 0.430 strict positives-only Dice (IoU 0.333, all-patch Dice 0.515, 105/141 positives de-

It.	Chars	Train	Test F1 (soft)		Dice
			pos	all	pos _{str}
0	49	0.156	0.013	0.210	0.012
13	98	0.172	0.000	0.193	0.000
16	77	0.172	0.000	0.199	0.000
21	85	0.172	0.000	0.193	0.000
24	41	0.555	0.563	0.587	0.444

Table 3: Full prompt frontier (soft-optimized run, one seed, *original ungrouped split*). Train = soft tolerance-F1 selection objective on the fixed 64-patch fitness sample; Test = held-out 176-patch test split (141 positive, 35 empty), soft-F1 ($\tau = 2$) positives-only and all-patch, plus strict pixel Dice positives-only. Champion success rate 87/141 at soft-F1 ≥ 0.5 ; 24/35 empty patches kept clean.

tected, only 5/35 empties clean—this earlier strict-scored pass over-fires on intact rock, unlike the later soft-metric run). The same terse phrase wins under both metrics, but strict Dice penalizes hair-thin cracks heavily—0.430 strict positives-only Dice vs. the soft champion’s 0.563 positives-only soft-F1—which is precisely the metric-ceiling effect the soft tolerance-F1 is designed to avoid.

SAM 3 inference. Open-weight SAM 3 (official release facebookresearch/sam3, transformers 5.0.0 Sam3Model/Sam3Processor), GPU-only, confidence threshold 0.5. Each prompt is a single concept query; returned instance masks are unioned, resized to annotation resolution, and binarized. Patches are pre-cut (full-image inference mode); an empty prediction on an empty patch scores Dice 1.0 and only a genuine inference error zeroes a sample. Throughput ≈ 6.5 s/patch on a GTX 1080 Ti (single-image, unbatched transformers inference); the distilled U-Net’s $\sim 1700\times$ speed-up is measured under the same single-patch serving regime, not against a batched or optimized SAM 3 server.

Matched-cap LoRA baseline (§6.2). Rank-8 LoRA (Hu et al., 2022) adapters ($\alpha = 16$, dropout 0.05, no bias term, 1.29 M trainable parameters) are inserted into the q_proj/v_proj projections of the vision backbone and of the DETR encoder and decoder attention modules; the text encoder, geometry encoder, mask/segmentation heads, and presence head stay frozen. Training uses all 96 patches of the adaptation cap at 1008^2 input with flips ($p = 0.5$) and mild brightness/contrast jitter ($p = 0.25$, limit 0.10; no rotation or cropping, which destroy

thin masks), an equally weighted BCE-with-logits + soft-Dice loss that keeps empty patches at full weight, AdamW (weight decay 0.01, cosine schedule, 10% warmup, gradient clipping 1.0), BF16, and a global effective batch of 8 for at most 40 epochs. The learning rate 3×10^{-5} was chosen from $\{10^{-5}, 3 \times 10^{-5}, 10^{-4}\}$ on grouped validation with a single development seed; checkpoints are evaluated every two epochs and selected by validation all-patch soft-F1 (ties broken by positives-only soft-F1, then empty-FP count, then earlier epoch), with early stopping after six evaluations without a 0.002 gain. The test split stays untouched until the configuration is frozen; the selected configuration is then re-run over seeds $\{0, 1, 2\}$ and evaluated once per seed. Prompt fitness uses a fixed 64-patch subset of the same cap while LoRA is conservatively allowed all 96, so the cap is matched but realized label access is not identical. Over three seeds LoRA reaches 0.487 ± 0.016 positives-only soft-F1, 0.371 ± 0.017 strict Dice, 0.519 all-patch soft-F1, and $14.0 \pm 1.7/38$ empty-patch false positives, against 0.545 ± 0.011 , 0.429 ± 0.012 , 0.589, and $9.7 \pm 1.2/38$ for the evolved prompt. We claim superiority over this configuration only, not over PEFT families, other ranks, or larger budgets. BF16 LoRA training ran on $2 \times$ NVIDIA A100 (40 GB), not on the GTX 1080 Ti nodes used for the frozen-segmentor and U-Net measurements.

Uncertainty estimation. Reported spreads are means $\pm$ SD over three independent seeds; fractional empty-FP counts are three-seed means over the 38 empty test patches. Paired differences resample the 23 test cores as clusters (10,000 bootstrap replicates, 95% level, each adapted system paired against the frozen evolved prompt), preserving within-core correlation rather than treating the 172 test patches as independent; the prompt-minus-LoRA difference is $+0.058$, 95% CI $[+0.020, +0.094]$. Comparisons across the ungrouped and grouped protocols are not paired, because test composition differs.

Input geometry matters at deployment. Raw drill-core strips are extreme-aspect images (median height/width ≈ 13.5 , up to 18; ~ 235 px wide, up to ~ 8000 px tall). Feeding a whole strip to SAM 3’s square input (1008^2) compresses it $\sim 13\times$ and erases hairline cracks—on a batch of 225 unlabeled strips, whole-strip inference returned a non-empty mask for only 54/225 (24%). Slicing instead into the training geometry recovers thin cracks with no

	Supervised	Ours
<i>Adaptation per new domain (where the arms differ)</i>		
New expert time	full corpus (~ 136 patches)	scoring probe (tens of patches)
Training GPU-hours	~ 0.05	~ 7 evolve + 0.6 distill
Adaptation bottleneck	annotation (days)	compute (hours)
<i>Deployed model (identical artifact)</i>		
Model size	7.76 M (31.1 MB)	
Inference VRAM	1.9 GB	
Latency / patch	3.8 ms GPU / 161 ms CPU	
Off-site exposure	none (on-prem)	

Table 4: Cost and data-locality (§6.7). Both arms deploy the same small U-Net; they differ only in per-domain adaptation. Ours trades GPU compute for expert annotation time. Inference on one GTX 1080 Ti.

pre/post-processing: a 2:1 height-tiling (overlap 15%, masks OR-merged) lifts non-empty coverage to 200/225 (89%). We use 2:1 tiling for full-core pre-labeling.

Evolution. MAP-Elites (Mouret and Clune, 2015) on GigaEvo (Khurlov et al., 2025); mutation LLM Qwen3-235B-A22B-Instruct-2507 on a local endpoint (max_tokens 16384, one parent per mutation); budget 2000 candidates (an initial 1000-candidate run extended to 2000); concurrency 4 with one candidate per GPU via a file-lock round-robin GPU selector. Validation uses *no* LLM: fitness is the mean soft tolerance-F1 over a fixed 64-patch subsample of the 96-patch train probe (Table 5), so scoring is deterministic and auditable. The ungrouped run starts from four seed prompts: “dark thin open fractures in geological drill core”, “black branching cracks and fissures in rock core”, “light mineral filled healed fractures in rock core”, and a longer “visible cracks . . . high-contrast hairline” phrase. The grouped re-evolution starts from eight seeds covering the same design space—geometry (“thin irregular fracture trace crossing drill core”), a refinement of the previous champion (“dark thin branching fracture in drill core”), core-interior (“narrow branching fracture within rock core material”) and dual-polarity (“dark open crack or pale mineral-filled fracture in drill core”) phrasings, a hairline-fissure and a pale-healed specialist, a branching-network variant, and one hard-negative phrasing (“thin branching fracture in drill core, not a shadow or core edge”)—and converges on the champion-refinement seed itself: “dark thin branching fracture in drill core”.

Splits and grouped-split provenance. The 450 patches are partitioned source-grouped into 99/179/172 from 18/25/23 cores (Table 5). The

Split	Cores	Patches	Notes
Train (cap)	18	99	83 pos; 96-cap, 3 unused
Validation	25	179	143 pos; source-disjoint
Test	23	172	134 pos, 38 empty
All	66	450	zero core overlap

Table 5: CoreFrac source-grouped splits, derived from 200 full-resolution images. Headline methods adapt from the 96-patch cap only.

assignment is deterministic: source-core groups are sorted by core id, shuffled with Python `random.Random(42)`, and assigned *whole* to train, then validation, then test until each patch target is met. Whole-group assignment overshoots the nominal 96/178 targets, giving realized sizes 99/179/172 (83+16, 143+36, 134+38 positive+empty). The fixed adaptation cap is the first 96 sorted training ids (80 positive, 16 empty); the three excluded patches are all positive and are not moved to another split. A checker verifies zero sample overlap, zero source-core overlap, and zero spatial overlap, that all 450 image/mask pairs exist with matching crop geometry, that masks are binary, and that labels agree with mask contents; the same checker reports 61/66 cores spanning more than one split in the original patch-level release. The builder, the split manifest, and the adaptation pool ship with the data, so the checker and its validation report (including the input-manifest SHA-256) can be regenerated from the release. The separate domain-shift split below is also source-aware but uses a brightness cut, not this headline assignment.

Domain-shift split (§6.3). For the maintenance study the 66 source cores are partitioned by source-mean patch brightness into domain A (33 lighter cores, brightness 0.527–0.676, healed fractures) and domain B (33 darker cores, brightness 0.221–0.524, open fractures), each with its own source-aware train/val/test split (A-test 71 patches = 56 positive + 15 empty; B-test 101 = 77 + 24). This keeps whole cores within a single domain so the A→B drop measures a genuine shift, not leakage. Per-domain prompts are evolved with the same protocol (MAX_MUTANTS=300 per domain); for the representative seed of Table 2, champion A is “cracks in drill core” and champion B “thin dark crack in core”, while seeds 2–3 converge on a different phrase (§6.3).

Distillation recipe. The deployable model is a compact U-Net (width 32, depth 4; 7.76 M pa-

Prompt (source)	Dice	Prec.	Succ.
“crack” (generic)	0.457	0.54	0.71
drill-core phrase (transferred as-is)	0.460	0.55	0.72
re-evolved on a 15-patch CRACK500 probe	0.471	0.57	0.75
over-specified phrase	0.135	0.15	0.21

Table 6: External-domain check on CRACK500 (1,124 pavement test patches, strict pixel Dice, default threshold; §6.4). A drill-core-evolved prompt transfers as well as a generic one, and the same language-only loop on a 15-patch probe adds a small further gain. Pavement cracks are easier than drill-core hairlines, so this is a transfer/adaptation demonstration, not a head-to-head accuracy comparison.

rameters, 31.1 MB checkpoint, 256×256 input) trained on the SAM 3 champion’s pseudo-labels over the 96-patch grouped train cap and scored on the 172-patch grouped test. No human labels enter the targets. The decision threshold (0.85) and the checkpoint are selected on grouped validation by all-patch soft-F1 (empty-aware), with `pos_weight` capped at 8. Inference cost (one GTX 1080 Ti): 3.82 ms/patch on GPU, 160.6 ms on CPU (4 threads), peak 1.88 GB VRAM.

Data pools (benchmark vs. deployment). CoreFrac is a curated, controlled benchmark (66 cores → 200 images → 450 patches). The 118 additional cores (new-data generalization) and the 1,163-patch pool (supervised scaling) below come from ongoing deployment annotation and are kept strictly separate from the benchmark with no source overlap.

Supervised data scaling. Training the same U-Net from scratch on an expanded set of newly annotated cores (1163 train patches vs. the 136 of the domain-B retrain baseline) lifts positives-only soft-F1 from 0.647 to 0.745 (strict 0.515 → 0.630) and roughly halves the empty-FP rate (58% → 39%). These are scored on the expanded set’s own held-out split, not Table 2 or the grouped test; the trend—cheap pre-labeled annotation improves the on-prem model—is the operative point.

New-data generalization at deployment resolution. To test the deployable model on genuinely unseen material, we ran the frozen distilled U-Net (no retraining) on 118 newly annotated full cores that share no source with the 66 cores the 450-patch benchmark was cut from. Cores are tall narrow strips (e.g. 254×3558 px); we infer at deployment resolution by cutting 2:1 vertical tiles (height = $2 \times$ width, 50% overlap) and unioning the per-

tile predictions—the same 2:1 height-tiling geometry remedy described above. The distilled U-Net holds 0.472 positives-only soft tolerance-F1 (strict 0.399, IoU 0.260) on the 118 unseen cores, *above* its 0.425 (strict 0.321) on the 66 source cores: the on-prem model does not degrade on new material. Both figures are full-core evaluations at deployment resolution and are therefore not comparable to the patch-level 0.501 of Table 1.

Best human-expert prompt baseline (§6.2).

Eight expert-written phrasings were scored under the identical soft metric ($\tau = 2$). The best, “clearly visible fractures in drill core”, reaches 0.212 positives-only soft-F1 (0.168 strict, success rate 23.4%, 1/35 empty false positives). The remaining seven score ≤ 0.053 positives-only: several (e.g. “fractures in drill core”, “fracture lines in rock core”) collapse to zero true positives on the held-out positives, while a verbose one (“mineralized or weathered natural fractures in rock core”) over-segments, firing on 30/35 empty patches. The evolved champion (0.563) thus outcores the best hand-crafted expert prompt by $2.7\times$ —the comparison that justifies automated search.

Deployment lessons. Four lessons generalize beyond this dataset. Square-input geometry is a trap: whole-strip SAM 3 coverage was 54/225 (24%), while 2:1 height-tiling lifted it to 200/225 (89%). More descriptive is not better: verbose prompts overfit the small probe. The operative tokens are generic (“open”, “crack”). Default distillation hyperparameters silently broke empty-patch behavior until threshold, pos_weight, and checkpoint selection were made empty-aware.

Zero-shot and open-vocabulary baselines (§6.2).

Their comparison target is the evolved champion’s 0.563 (Table 3). The zero-shot VLM row is the best of four API VLMs prompted to trace fractures as normalized polylines rasterized to masks: Qwen3-VL-30B reaches 0.103 positives-only soft-F1 and Qwen3-VL-235B 0.043; two reasoning VLMs (Gemini-2.5-pro, GPT-5.5) returned no parseable polyline under the JSON protocol and are omitted. Even the best is $5.5\times$ below the evolved frozen-SAM 3 prompt—asking a VLM to “draw the mask” does not work for hairline fractures. The open-weight on-prem segmentors run locally with no API: OWLv2→SAM is open-vocabulary detection (noun prompt “crack”) feeding SAM boxes, and CLIPSeg is text→dense-mask with the

decision threshold calibrated on validation. The best open zero-shot model (OWLv2→SAM, 0.211) only ties the best hand-written SAM 3 prompt and stays $2.7\times$ below the evolved prompt while over-firing on empty patches (18/35); for CLIPSeg the evolved descriptive phrase lifts soft-F1 from 0.020 (bare “crack”) to 0.152, so prompt search helps beyond SAM 3. Florence-2 returns a polygon for every patch and hallucinates a crack on all 35 empty patches. The headline evolved champion is one seed’s; an independent second seed re-derived the identical champion on the train objective (0.5547, matching to four decimals), and per-seed held-out variance is reported for the maintenance study (§6.3).

Measurements not run. Two measurements lie outside the reported evidence and remain future work: the counterbalanced annotation-timing protocol (matched from-scratch vs. correction sets with a shared timing tool); and the expert-corrected and same-minute-budget distillation controls (§6.6).

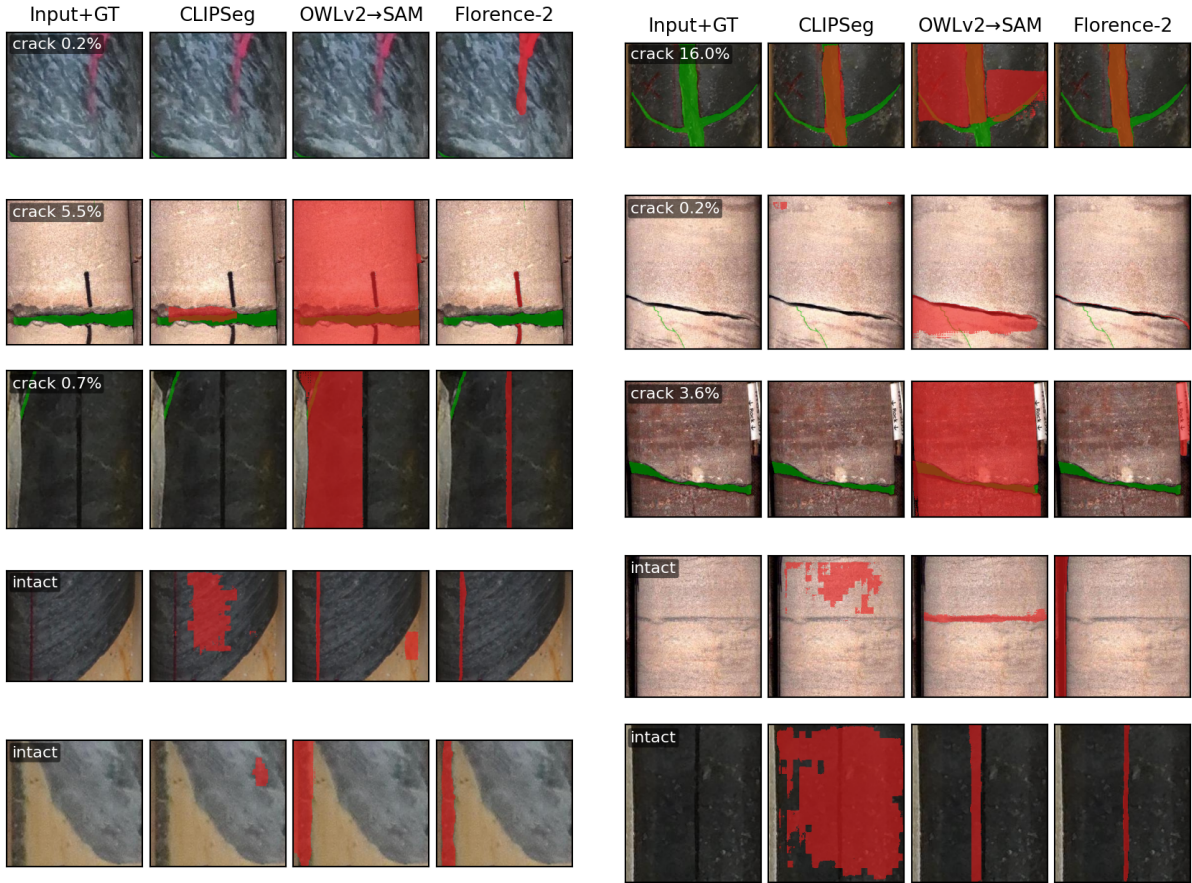


Figure 4: Failure modes of open-weight zero-shot segmentors (prediction in red, expert GT in green; best viewed in color—predictions and GT occupy different regions here, so they remain distinguishable in grayscale). **Top four rows (intact patches, empty GT):** all three models hallucinate fractures on plain rock—CLIPSeg paints large blobs over rock texture, while OWLv2→SAM and Florence-2 lock onto vertical depth-marker strokes and core-tray edges, exactly the *structured* non-rock content that a deployable detector must ignore. **Bottom row (hairline crack):** the true crack is a faint green hairline; all three fire elsewhere and miss it. This is why open zero-shot models over-fire on empty patches (Table 1) yet stay far below the evolved frozen-SAM 3 prompt.